Custom Recorders

Documentation by Alex Baker, 2021

Output in OpenSees is either handled by the OpenSees recorder command, which records model information at each successful analysis step, by printing to screen (and to file if the OpenSees command logFile is used) with the Tcl puts command, or by outputting directly to a file, using the Tcl commands open, puts, and close, to open, print to, and close the file, respectively. However, handling file identifiers with the Tcl commands open and close can get complicated, especially if the user wants to print to multiple files.

This file (CustomRecorders.tcl) is intended to simplify this process by providing custom recorder classes in OpenSees/Tcl through use of command traces and command wrapping. Two new recorder classes are added: "Custom" and "EnvelopeCustom". Both custom classes evaluate code in the global namespace (in the order in which the recorders are created) and write the result to file upon every successful analysis step and every invocation of "record".

recorder Custom <-file $filename> <-fileAdd $filename> <-time> <-precision $nSD> <-format $fmt> $body

recorder EnvelopeCustom <-file $filename> <-fileAdd $filename> <-time> <-precision $nSD> <-format $fmt> $body

$filename	Filename to print output to (required). Overwrites with option -file, appends with option -fileAdd.
-time	Option to print analysis time (from getTime) as first column.
$nSD	Number of significant digits to print output to. Default 6.
$fmt	Format to print output values with (e.g. "%s", "%.3e"). Overrides -precision
$body	Script to evaluate in global namespace. If a "return" statement is used, it will print the return value.

Functionality of the custom recorders was designed to replicate the functionality of normal recorders as closely as possible.

- When a custom recorder is created, it opens a file for writing, and then writes lines to the file every successful analysis step and/or call of the "record" command.
- All custom recorders can be closed/removed with "wipe" or "remove recorders".
- Individual recorders can be closed/removed with "remove recorder $recorderTag", where the recorder tag is the return value of the recorder command. Normal recorder tags start at 0 and increment, custom recorder tags start at -1 and decrement.
- Recorder data can be accessed during the analysis with the "recorderValue" command, using the corresponding recorder tag. Functionality of the "recorderValue" command is not well documented, but custom recorders should behave the same as normal recorders for this.